PATENT COOPERATION TREATY

From the
INTERNATIONAL PRELIMINARY EXAMINING AUTHORITY

To:

MARGOLIS, Andrew
Legal Department
Symbian Software Limited
2-6 Boundary Row
London SE1 8HP
GRANDE BRETAGNE

PCT

NOTIFICATION OF TRANSMITTAL OF THE INTERNATIONAL PRELIMINARY REPORT ON PATENTABILITY

(PCT Rule 71.1)

Date of mailing
(day/month/year) 26.04.2006

Applicant's or agent's file reference

IMPORTANT NOTIFICATION

International application No. PCT/GB2005/001300	International filing date *(day/month/year)* 01.04.2005	Priority date *(day/month/year)* 02.04.2004

Applicant
SYMBIAN SOFTWARE LIMITED

1. The applicant is hereby notified that this International Preliminary Examining Authority transmits herewith the international preliminary report on patentability and its annexes, if any, established on the international application.

2. A copy of the report and its annexes, if any, is being transmitted to the International Bureau for communication to all the elected Offices.

3. Where required by any of the elected Offices, the International Bureau will prepare an English translation of the report (but not of any annexes) and will transmit such translation to those Offices.

4. **REMINDER**

The applicant must enter the national phase before each elected Office by performing certain acts (filing translations and paying national fees) within 30 months from the priority date (or later in some Offices) (Article 39(1)) (see also the reminder sent by the International Bureau with Form PCT/IB/301).

Where a translation of the international application must be furnished to an elected Office, that translation must contain a translation of any annexes to the international preliminary report on patentability. It is the applicant's responsibility to prepare and furnish such translation directly to each elected Office concerned.

For further details on the applicable time limits and requirements of the elected Offices, see Volume II of the PCT Applicant's Guide.

The applicant's attention is drawn to Article 33(5), which provides that the criteria of novelty, inventive step and industrial applicability described in Article 33(2) to (4) merely serve the purposes of international preliminary examination and that "any Contracting State may apply additional or different criteria for the purposes of deciding whether, in that State, the claimed inventions is patentable or not" (see also Article 27(5)). Such additional criteria may relate, for example, to exemptions from patentability, requirements for enabling disclosure, clarity and support for the claims.

Name and mailing address of the international preliminary examining authority:

European Patent Office - P.B. 5818 Patentlaan 2
NL-2280 HV Rijswijk - Pays Bas
Tel. +31 70 340 - 2040 Tx: 31 651 epo nl
Fax: +31 70 340 - 3016

Authorized Officer

Bota-Madsen, Z

Tel. +31 70 340-2393

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY REPORT ON PATENTABILITY

(Chapter II of the Patent Cooperation Treaty)

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference	**FOR FURTHER ACTION**	See Form PCT/IPEA/416
International application No. PCT/GB2005/001300	International filing date *(day/month/year)* 01.04.2005	Priority date *(day/month/year)* 02.04.2004
International Patent Classification (IPC) or national classification and IPC INV. G06F9/46		
Applicant SYMBIAN SOFTWARE LIMITED		

1. This report is the international preliminary examination report, established by this International Preliminary Examining Authority under Article 35 and transmitted to the applicant according to Article 36.
2. This REPORT consists of a total of 5 sheets, including this cover sheet.
3. This report is also accompanied by ANNEXES, comprising:

   a. ☒ *sent to the applicant and to the International Bureau)* a total of 8 sheets, as follows:

   ☒ sheets of the description, claims and/or drawings which have been amended and are the basis of this report and/or sheets containing rectifications authorized by this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions).

   □ sheets which supersede earlier sheets, but which this Authority considers contain an amendment that goes beyond the disclosure in the international application as filed, as indicated in item 4 of Box No. I and the Supplemental Box.

   b. □ *(sent to the International Bureau only)* a total of (indicate type and number of electronic carrier(s)) , containing a sequence listing and/or tables related thereto, in electronic form only, as indicated in the Supplemental Box Relating to Sequence Listing (see Section 802 of the Administrative Instructions).

4. This report contains indications relating to the following items:

☒	Box No. I	Basis of the report
□	Box No. II	Priority
□	Box No. III	Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
□	Box No. IV	Lack of unity of invention
☒	Box No. V	Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
□	Box No. VI	Certain documents cited
□	Box No. VII	Certain defects in the international application
□	Box No. VIII	Certain observations on the international application

Date of submission of the demand 01.02.2006	Date of completion of this report 26.04.2006
Name and mailing address of the international preliminary examining authority: European Patent Office - P.B. 5818 Patentlaan 2 NL-2280 HV Rijswijk - Pays Bas Tel. +31 70 340 - 2040 Tx: 31 651 epo nl Fax: +31 70 340 - 3016	Authorized officer de Man, A Telephone No. +31 70 340-4527 Europäisches Patentamt · European Patent Office · Office européen des brevets

Box No. I Basis of the report

1. With regard to the **language**, this report is based on

☒ the international application in the language in which it was filed

☐ a translation of the international application into , which is the language of a translation furnished for the purposes of:
   ☐ international search (under Rules 12.3(a) and 23.1(b))
   ☐ publication of the international application (under Rule 12.4(a))
   ☐ international preliminary examination (under Rules 55.2(a) and/or 55.3(a))

2. With regard to the **elements*** of the international application, this report is based on *(replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report)*:

Description, Pages

1-5, 8-16	as originally filed
6, 7, 7a	received on 02.02.2006 with letter of 31.01.2006

Claims, Numbers

1-31	received on 02.02.2006 with letter of 31.01.2006

Drawings, Sheets

1/1	as originally filed

☐ a sequence listing and/or any related table(s) - see Supplemental Box Relating to Sequence Listing

3. ☐ The amendments have resulted in the cancellation of:
   ☐ the description, pages
   ☐ the claims, Nos.
   ☐ the drawings, sheets/figs
   ☐ the sequence listing *(specify)*:
   ☐ any table(s) related to sequence listing *(specify)*:

4. ☐ This report has been established as if (some of) the amendments annexed to this report and listed below had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).
   ☐ the description, pages
   ☐ the claims, Nos.
   ☐ the drawings, sheets/figs
   ☐ the sequence listing *(specify)*:
   ☐ any table(s) related to sequence listing *(specify)*:

* *If item 4 applies, some or all of these sheets may be marked "superseded."*

INTERNATIONAL PRELIMINARY REPORT ON PATENTABILITY

International application No.
PCT/GB2005/001300

Box No. V Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-31
	No:	Claims	
Inventive step (IS)	Yes:	Claims	
	No:	Claims	1-31
Industrial applicability (IA)	Yes:	Claims	1-31
	No:	Claims	

2. Citations and explanations (Rule 70.7):

see separate sheet

Re Item V

Reasoned statement with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1 Reference is made to the following document:

D1: Cherepov, "Hard Real-Time With RTX on Windows NT" (July 1999)

2 Document D1 discloses, using the wording of claim 1 insofar as possible,

a computing device comprising a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities (page 107, right-hand column, lines 8-19; RTSS scheduler)*; and*

wherein a ready list is kept of threads which are scheduled to run on the device, ordered by priority (page 107, right-hand column, lines 20-30; ready queue per priority)*;*

the device further comprising at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism (page 107, right-hand column, lines 43-49; access to an object is blocked for a high priority thread while a low priority thread holds the object)*;*

and wherein, when *a scheduled thread is blocked from running because the resource it requires is locked, the thread which holds the lock is caused to run* (page 107, right-hand column, lines 43-49; the effective priority of the low priority thread that holds the object is promoted to that of the high priority thread, thereby implicitly causing it to run),

from which the subject-matter of claim 1 differs in that the blocked thread is not removed from its place on the ready list.

This difference merely relates to an implementation detail of the priority inheritance scheme. The fact that the blocked thread is not removed from the ready list does not by itself imply any technical advantage over what is disclosed in document D1. Not removing a blocked thread from the ready list, but, for example, flagging it as blocked, is therefore regarded as an obvious design possibility for the skilled person.

Thus the subject-matter of claim 1 does not involve an inventive step (Article 33(3) PCT).

3 The subject-matter of independent claims 17 and 18 largely corresponds to the subject-matter of claim 1, which is found to lack inventive step. Thus, for substantially the same reasons as above, the subject-matter of claims 17 and 18 is not new either (Article 33(3) PCT).

4 The additional features of dependent claims 2, 4, 19 and 21 are also disclosed by document D1, see page 107, right-hand column, lines 8-30.

Thus the subject-matter of claims 2, 4, 19 and 21 does not involve an inventive step (Article 33(3) PCT).

5 Dependent claims 3, 5-16, 20 and 22-31 specify obvious and well-known features that merely define implementation details of the scheduler and locking mechanism, not affecting the actual scheduling scheme or achieving any non-obvious technical advantages. These features are therefore not considered to contribute to an inventive step.

Thus the subject-matter of claims 3, 5-16, 20 and 22-31 does not involve an inventive step (Article 33(3) PCT).

A deficiency of many current microkernel designs is that they feature separate and computationally expensive priority inheritance mechanisms. Hence, turning them off to boost operating system performance is always going to be a temptation. An architecture which provides automatic priority inheritance without any performance penalty is therefore preferable to existing designs, since it would not force robustness and performance to be traded off against each other. This is especially true where the nanokernel is used to host one or more operating systems, since any unreliability in the nanokernel translates directly into unreliability in the operating system as a whole.

It is therefore an object of the present invention to provide an improved form of computing device which is able to display automatic priority inheritance without any performance penalty.

According to a first aspect of the present invention there is provided a computing device comprising a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities; and including a list of threads which are scheduled to run on the device, ordered by priority; the device further comprising at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and in which a scheduled thread which is blocked from running causes the thread which holds the locking mechanism to run.

According to a second aspect of the invention there is provided a method of operating a computing device, the method comprising providing a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities, and including a ready list of threads which are scheduled to run on the device, ordered by priority; providing at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and arranging for a scheduled thread which is blocked from running because the resource it requires is locked not to be removed from its place on the ready list and instead the thread which holds the locking mechanism is caused to run.

According to a third aspect of the present invention there is provided an operating system for a computing device, the operating system comprising a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities, and including means for providing a ready list of threads which are scheduled to run on the device, ordered by priority; at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and means for arranging for a scheduled thread which is blocked from running because the resource it requires is locked not to be removed from its place on the ready list and instead the thread which holds the locking mechanism to caused to run.

An embodiment of the present invention will now be described, by way of further example only, with reference to the accompanying drawing which illustrates examples of monolithic and micro kernel architectures.

The present invention will be described with specific reference to the Symbian OS^{TM} operating system available from Symbian Limited of London, England. However, it is to be understood that the principles of the present invention may also be used to equal advantage in other types of operating system.

The Symbian OS operating system includes a pre-emptive multi-threaded nanokernel style layer providing hard real-time support. It is within this nanokernel that an implementation of a single mechanism which is referred to as a *fast mutex* is to be found. This combines the benefits of a method for kernel threads to provide mutual exclusion locks on resources with the benefits of a method for automatic priority inheritance for the task holding the mutex. The multi-threaded nature of the kernel makes it suitable for use in either single-processor or symmetrical multiprocessing (SMP) systems, and the fast mutex can be used in either configuration.

It is not considered necessary to fully describe the Symbian OS operating system in order to provide a sufficient understanding of this invention. Thus, the following description is restricted to those parts of the operating system relevant to the invention. The Symbian OS kernel is a hybrid between the monolithic and microkernel approaches shown in figure 1, and therefore

CLAIMS

1. A computing device comprising a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities; and wherein a ready list is kept of threads which are scheduled to run on the device, ordered by priority; the device further comprising at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and wherein a scheduled thread which is blocked from running because the resource it requires is locked is not removed from its place on the ready list and instead the thread which holds the locking mechanism is caused to run.

2. A computing device according to claim 1 wherein states are assigned to threads and the list comprises of all threads having a common state.

3. A computing device according to claim 2 wherein a blocked thread is not permitted to change its state.

4. A computing device according to any one of claims 1 to 3 wherein the list is subdivided in accordance with the priority of the threads it contains.

5. A computing device according to any one of the preceding claims wherein a thread is arranged to contain a pointer to any locking mechanism it is blocked on.

6. A computing device according to any one of the preceding claims comprising a plurality of non-nestable locking mechanisms.

7. A computing device according to any one of the preceding claims wherein the scheduler is arranged to be called at the end of an interrupt service routine which is caused to run on the computing device.

8. A computing device according to any one of the preceding claims wherein the locking mechanism(s) comprise(s) a mutex including a pointer, which is null if the mutex is free or points to the thread holding the mutex, and includes a flag indicating whether or not the mutex is contested.

9. A computing device according to claim 8 wherein the algorithm is arranged to delegate memory management to a replaceable memory model configured in dependence upon the configuration of the computing device.

10. A computing device according to claim 9 wherein the memory model is arranged to run in either pre-emptible or non-preemptible modes.

11. A computing device according to claim 10 wherein a mutex is arranged to protect the module from running in the pre-emptible mode.

12. A computing device according to any one of the preceding claims wherein the scheduler is included in a kernel of an operating system of the computing device.

13. A computing device according to claim 12 wherein the kernel comprises a microkernel or a nanokernel and where the threads are, respectively, microkernel or nanokernel threads.

14. A computing device according to claim 12 or 13 wherein the scheduler is arranged to be called each time the kernel is unlocked.

15. A computing device according to any one of the preceding claims comprising a mobile computing device.

16. A computing device according to claim 15 comprising a smart phone.

17. A method of operating a computing device, the method comprising providing a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities, and including a ready list of threads which are scheduled to run on the device, ordered by priority; providing at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and arranging for a scheduled thread which is blocked from running because the resource it requires is locked not to be removed from its place on the ready list and instead the thread which holds the locking mechanism is caused to run.

18. An operating system for a computing device, the operating system comprising a scheduler incorporating an algorithm for ordering the running of threads of execution having different priorities, and including means for providing a ready list of threads which are scheduled to run on the device, ordered by priority; at least one locking mechanism for blocking access to a resource of the device from all threads except for a thread that holds the locking mechanism; and means for arranging for a scheduled thread which is blocked from running because the resource it requires is locked not to be removed from its place on the ready list and instead the thread which holds the locking mechanism to caused to run.

19. An operating system according to claim 18 wherein states are assigned to threads and the list comprises of all threads having a common state.

20. An operating system according to claim 19 arranged to inhibit a blocked thread from changing its state.

21. An operating system according to any one of claims 18 to 20 arranged to subdivide the list in accordance with the priority of the threads it contains.

22. An operating system according to any one of claims 18 to 21 wherein a thread is arranged to contain a pointer to any locking mechanism it is blocked on.

23. An operating system according to any one claims 18 to 22 comprising a plurality of non-nestable locking mechanisms.

24. An operating system according to any one of claims 18 to 23 wherein the scheduler is arranged to be called at the end of an interrupt service routine which is caused to run on the computing device.

25. An operating system according to any one of claims 18 to 24 wherein the locking mechanism(s) comprise(s) a mutex including a pointer, which is null if the mutex is free or points to the thread holding the mutex, and includes a flag indicating whether or not the mutex is contested.

26. An operating system according to claim 25 wherein the algorithm is arranged to delegate memory management to a replaceable memory model configured in dependence upon the configuration of the computing device.

27. An operating system according to claim 26 wherein the memory model is arranged to run in either pre-emptible or non-preemptible modes.

28. An operating system according to claim 27 wherein a mutex is arranged to protect the module from running in the pre-emptible mode.

29. An operating system according to any one of claims 18 to 28 wherein the scheduler is included in the kernel.

30. An operating system according to claim 29 wherein the kernel comprises a microkernel or a nanokernel and where the threads are, respectively, microkernel or nanokernel threads.

31. An operating system according to claim 29 or 30 wherein the scheduler is arranged to be called each time the kernel is unlocked.